CLAIMS

1. A method comprising:

minting a stick of electronic assets by digitally signing with an issuer's signature a composite of user-provided data items including a user identity, a bottom asset from a bottom of the stick, and a length of the stick;

spending one or more assets from the stick at one or more vendors, wherein each expenditure with a particular vendor involves digitally signing with a user's signature a first asset from the stick to be spent and passing the user-signed first asset along with the issuer-signed composite to the particular vendor for verification and subsequently passing any additional assets to be spent without user signature to the particular vendor; and

depositing one or more assets collected by the particular vendor by digitally signing with the particular vendor's signature a composite of data items including the user-signed first asset and a last asset spent by the user from the stick and passing the vendor-signed composite along with the issuer-signed composite to the issuer.

2. A method as recited in claim 1, further comprising storing the stick of electronic assets in a tamper-resistant electronic wallet.

3. A method as recited in claim 1, further comprising storing the stick of electronic assets in an electronic wallet constructed with a secure-processor architecture.

4. A method as recited in claim 1, wherein the minting comprises minting the stick of assets using a blind signature protocol.

5. A method as recited in claim 1, wherein the spending comprises:

concatenating a vendor identity with the first asset from the stick to form a payment request;

signing the payment request with a signature of the user:

submitting the user-signed payment request along with the issuer-signed withdrawal request to the vendor;

accepting the first asset as payment in an event that the user and the issuer are verified; and

subsequently passing any additional assets from the stick as payment to the vendor without digitally signing them with the user's signature;

6. A method as recited in claim 1, wherein the depositing comprises:

concatenating the user-signed first asset $S_U(Cj)$, a last asset spent from the stick *Ck*, and a run length *RL* of assets beginning with the first asset *Cj* and ending with the last asset *Ck* to form a deposit request;

signing the deposit request with a signature of the vendor:

$$S_V(S_U(Cj), Ck, RL)$$

submitting the vendor-signed deposit request along with the issuer-signed withdrawal request to the issuer; and

crediting a vendor account with the run of assets in an event that the user, the vendor, the run, and the issuer are positively verified.

7. A method as recited in claim 1, further comprising auditing the assets deposited by the vendor.

8. A method as recited in claim 1, further comprising auditing a sample of the assets paid by the user to the vendor.

9. A method as recited in claim 1, further comprising selecting, at the vendor, a subset of less than all of the assets paid by the user to the vendor and submitting the subset of assets to an auditor for fraud evaluation.

10. Distributed computer-readable media resident at the issuer, user, and vendor having computer-executable instructions to perform the method as recited in claim 1.

11. Computers resident at the issuer, user, and vendor that are programmed to perform the method as recited in claim 1.

12. A method for issuing electronic assets, comprising:

forming a stick of L electronic assets C_i (for i=1, …, L) where each asset can be derived from a preceding asset in the stick;

signing the stick with a signature of a party issuing the assets;

spending a first run of one or more assets from the stick at a first vendor; and

spending a second run of one or more assets from the stick at a second vendor.

13. A method as recited in claim 12, further comprising storing the stick of electronic assets in a tamper-resistant electronic wallet.

14. A method as recited in claim 12, further comprising storing the stick of electronic assets in an electronic wallet constructed with a secure-processor architecture.

15. A method as recited in claim 12, wherein the forming comprises anonymously issuing the stick of assets using a blind signature protocol.

16. A method as recited in claim 12, wherein the forming comprises:

creating the stick of L electronic assets by computing:

$$C_i = h^i(x) \quad (\text{for } i=1, \ldots, L)$$

where $h(x)$ is a one-way hashing function of a value x.

17. A method as recited in claim 16, wherein the forming further comprises:

constructing a withdrawal request having a user identity U, a user secret K, a last asset value C_L taken from a bottom of the stick, a denomination d indicating a value for the assets in the stick, an expiration t, and the value L; and

signing the withdrawal request with a signature of an issuer:

$$S_I(U, K, d, C_L, t, L).$$

18. A method as recited in claim 12, wherein the spending comprises:

signing a first asset from the stick with a signature of the user:

submitting the user-signed asset along with the signed stick to the first vendor; and

in an event the first asset is accepted, subsequently submitting any additional assets from the stick without digitally signing them.

19. A method as recited in claim 12, further comprising auditing the assets from the first and second runs of assets for fraud.

20. A method as recited in claim 12, further comprising auditing a sample of assets from the first and second runs of assets for fraud.

21. A method as recited in claim 12, further comprising depositing the first and second runs of assets.

22. Computer-readable media resident at the issuer and the user having computer-executable instructions to perform the method as recited in claim 12.

23. Computers resident at the issuer and the user that are programmed to perform the method as recited in claim 12.

24. A method for issuing electronic assets, comprising:

creating, at a user, a stick of L electronic assets by computing:

$$C_i = h^i(x) \quad (\text{for } i=1, \ldots, L)$$

where $h(x)$ is a hashing function of a value x;

submitting a withdrawal request from the user to an issuer, the withdrawal request having a user identity U, a last asset value C_L taken from a bottom of the stick, and the value L, while omitting any vendor identity;

signing, at the issuer, the withdrawal request; and

returning the signed withdrawal request to the user.

25. A method as recited in claim 24, further comprising storing the stick of electronic assets and signed withdrawal request in a tamper-resistant electronic wallet.

26. A method as recited in claim 24, further comprising storing the stick of electronic assets and signed withdrawal request in an electronic wallet constructed with a secure-processor architecture.

27. A method as recited in claim 24, wherein the withdrawal request further has a user secret K, a denomination d indicating a value for the assets in the stick, and an expiration t.

28. A computer-readable medium having computer-executable instructions that direct an electronic wallet to perform the method as recited in claim 24.

29. A computer programmed to perform the method as recited in claim 24.

30. A computer-readable medium storing the stick of electronic coins and the signed withdrawal request constructed as a result of the method as recited in claim 24.

31. A method comprising:

creating, at a user, a stick of L electronic assets by computing:

$$C_i = h^i(x) \quad (\text{for } i=1, \ldots, L)$$

where $h(x)$ is a hashing function of a value x;

submitting a withdrawal request from the user to an issuer, the withdrawal request having a user identity U, a user secret K, a last asset value C_L taken from a bottom of the stick, a denomination d indicating a value for the assets in the stick, an expiration t, and the value L;

signing, at the issuer, the withdrawal request:

$$S_I(U, K, d, C_L, t, L)$$

returning the issuer-signed withdrawal request to the user;

initiating payment of one or more assets from the stick to a vendor having an identity V;

concatenating, at the user, the vendor identity with a first asset Cj to be spent from the stick to form a payment request, and a depth D indicating a distance of the first asset from the bottom of the stick;

signing the payment request with a signature of the user:

$$S_U(Cj, D, V1)$$

submitting the user-signed payment request along with the issuer-signed withdrawal request to the vendor;

accepting the first asset as payment at the vendor in an event that the user and the issuer are verified;

subsequently passing any additional assets from the stick as payment to the vendor without digitally signing them with the user's signature;

concatenating, at the vendor, the user-signed first asset, a last asset spent from the stick *Ck*, and a run length *RL* of assets beginning with the first asset *Cj* and ending with the last asset *Ck* to form a deposit request;

signing the deposit request with a signature of the vendor:

$$S_V(S_U(C_j), Ck, RL)$$

submitting the vendor-signed deposit request along with the issuer-signed withdrawal request to the issuer; and

crediting a vendor account with the run of assets in an event that the user, the vendor, and the issuer are verified.

32. A method as recited in claim 31, further comprising randomly selecting an asset from the assets paid by the user to the vendor and submitting the selected asset for audit.

33. A method as recited in claim 31, further comprising auditing the assets deposited by the vendor with the issuer.

34. A method for anonymously issuing electronic assets, comprising:

creating, at a user, a stick of *L* electronic assets by computing:

$$C_i = h^i(x) \quad (\text{for } i=1, \ldots, L)$$

where *h(x)* is a hashing function of a value *x*;

blinding the stick using a random value p, where:

$$\text{Blind Stick} = p^e \, C_L \, mod \, N$$

where C_L is a bottom asset on the stick;

submitting a withdrawal request from the user to an issuer, the withdrawal request having the blind stick and the value L;

signing, at the issuer, the withdrawal request by computing:

$$c = (p^e C_L)^{Lf} = p^L C_L^{Lf} \, mod \, N$$

where e and f are public and private variables known by the issuer and e is known to everyone;

returning the signed withdrawal request to the user;

deriving a new bottom asset by computing:

$$C_L^{Lf} = c/p^L \, mod \, N.$$

35. A method as recited in claim 34, further comprising storing the blind stick of electronic assets and signed withdrawal request in a tamper-resistant electronic wallet.

36. A method as recited in claim 34, further comprising verifying the bottom asset by computing C_L^{Lf} independently and comparing a result to the new bottom asset derived in said deriving (C_L^{Lf}).

37. A method as recited in claim 34, further comprising storing the blind stick of electronic assets and signed withdrawal request in an electronic wallet constructed with a secure-processor architecture.

38. A method as recited in claim 34, further comprising spending an asset from the blind stick by first sending the new bottom to a vendor for verification.

39. A computer-readable medium having computer-executable instructions that direct an electronic wallet to perform the method as recited in claim 34.

40. A computer programmed to perform the method as recited in claim 34.

41. A computer-readable medium storing the blind stick of electronic coins and the signed withdrawal request constructed as a result of the method as recited in claim 34.

42. A method for handling electronic coupons, comprising:

creating dual sticks of corresponding coupons including a user stick located at the user and a vendor stick located at the vendor;

referencing at least one coupon in the user stick and at least one corresponding coupon in the vendor stick;

upon granting or spending a coupon, changing reference to a different coupon in the user stick and a different corresponding coupon in the vendor stick; and

swapping information between the user and the vendor to verify that the coupons being referenced in the user stick and the vendor stick correspond to one another.

43. A method as recited in claim 42, wherein the referencing comprises:

using a first user pointer to reference a newest coupon in the user stick and a second user pointer to reference an oldest unused coupon in the user stick; and

using a first vendor pointer to reference a newest coupon in the vendor stick and a second vendor pointer to reference an oldest unused coupon in the vendor stick.

44. A method as recited in claim 43, wherein the changing comprises, upon granting a newer coupon, moving the first user pointer and the first vendor pointer to reference the newer coupon.

45. A method as recited in claim 43, wherein the changing comprises, upon spending the oldest coupon, moving the second user pointer and the second vendor pointer to reference a next oldest coupon.

46. A method as recited in claim 42, wherein the swapping comprises exchanging data indicating which coupons are being referenced in the user stick and the vendor stick.

47. Computer-readable media resident at the user and the vendor having computer-executable instructions that perform the method as recited in claim 42.

48. Computers resident at the user and the vendor that are programmed to perform the method as recited in claim 42.

49. A method for handling electronic coupons, comprising:

storing a first set of coupons in a user-based data structure maintained at a user;

storing a second set of coupons in a vendor-based data structure maintained at a vendor, the second set of coupons corresponding to the first set of coupons;

using first and second user pointers to reference a first and last coupon in the user-based data structure;

using first and second vendor pointers to reference a first and last coupon in the vendor-based data structure;

upon earning a new coupon,

adding the new coupon to the user-based stick;

modifying the first user pointer at the user-based data structure to reflect the new coupon;

informing the vendor;

updating the first vendor pointer at the vendor-based data structure to reflect that the user-based data structure is referencing the new coupon;

upon spending a current coupon from the user-based data structure,

submitting the current coupon to the vendor;

evaluating, at the vendor, whether the coupon is acceptable and if acceptable, modifying the second pointer at the vendor-based data structure to reflect expenditure of the coupon;

informing the user;

updating the second user pointer at the user-based data structure to reflect expenditure of the current coupon.

50. An architecture for managing electronic coupons, comprising:

a user-based data structure embodied on a computer-readable medium, the user-based data structure storing one or more coupons;

a first user pointer to an oldest coupon in the user-based data structure;

a second user pointer to a newest coupon in the user-based data structure;

a vendor-based data structure embodied on a computer-readable medium, the vendor-based data structure storing one or more coupons associated with the coupons stored on the user-based data structure;

a first vendor pointer to an oldest coupon in the vendor-based data structure;

a second vendor pointer to a newest coupon in the vendor-based data structure;

wherein the user-based data structure and the vendor-based data structure are concurrently maintained so that (1) modification of the first user pointer to reference another coupon in the user-based stick results in updating and verification of the first vendor pointer to reference an associated coupon in the vendor-based stick and (2) modification of the second vendor pointer to reference a different coupon in the vendor-based stick results in updating and verification of the second user pointer to reference an associated coupon in the user-based stick.

251. An electronic asset system comprising:

an issuer wallet having a processor and storage, the issuer wallet digitally signing with an issuer's signature a composite of user-provided data items including a user identity, a bottom asset from a bottom of a stick of electronic assets, and a length of the stick;

a user wallet having a processor and storage to store the stick of electronic assets and issuer-signed composite and to spend one or more assets from the stick at one or more vendors, the user wallet spending one or more assets by digitally signing with a user's signature a first asset from the stick to be spent and passing the user-signed first asset along with the issuer-signed composite to the vendor for verification; whereupon verification, the user wallet subsequently passes any additional assets to be spent without user signature to the vendor; and

a vendor wallet having a processor and storage to store one or more assets spent by the user wallet, the vendor wallet depositing the assets collected from the user wallet by digitally signing with the particular vendor's signature a composite

of data items including the user-signed first asset and a last asset passed in the stick received from the user wallet and passing the vendor-signed composite along with the issuer-signed composite to the issuer wallet for verification.

52. An electronic asset system as recited in claim 51, wherein the issuer wallet, the user wallet, and the vendor wallet are tamper-resistant.

53. An electronic asset system as recited in claim 51, wherein the issuer wallet, the user wallet, and the vendor wallet are tamper-resistant constructed with a secure-processor architecture.

54. An electronic asset system as recited in claim 51, wherein the issuer wallet signs the composite using a blind signature protocol.

55. An electronic asset system as recited in claim 51, further comprising an auditing system to audit the electronic assets to detect whether assets have been used in a fraudulent manner.

56. An electronic asset system as recited in claim 51, further comprising a probabilistic auditing system to sample a subset of less than all electronic assets to detect whether assets have been used in a fraudulent manner.

57. An electronic wallet having memory and a processor, the electronic wallet being programmed to:

create a stick of L electronic assets by computing:

$$C_i = h^i(x) \quad (\text{for } i=1, \ldots, L)$$

where $h(x)$ is a hashing function of a value x;

form a withdrawal request having a user identity U, a last asset value C_L taken from a bottom of the stick, and the value L, while omitting any vendor identity;

submit withdrawal request to an issuer and receive the withdrawal request back with an issuer signature; and

store the signed withdrawal request and the stick.

58. An electronic wallet as recited in claim 57, further programmed to:

form a payment request for payment of one or more assets from the stick to a vendor having an identity V, the payment request having the vendor identity V and a first asset Cj to be spent from the stick;

sign the payment request:

$$S_U(Cj, V1)$$

submit the signed payment request along with the signed withdrawal request to the vendor.

59. An electronic wallet having memory and a processor, the electronic wallet being programmed to:

receive a run of assets from a user;

select a subset of less than all of the assets received from the user; and

submit the subset of assets to an auditor for evaluation of fraudulent expenditure.

60. An electronic wallet as recited in claim 59, further programmed to randomly select the subset of assets.